



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/842,527	04/25/2001	Christopher L. Anderson	MS158543.1	7282
27195	7590	08/24/2006		
AMIN. TUROCY & CALVIN, LLP 24TH FLOOR, NATIONAL CITY CENTER 1900 EAST NINTH STREET CLEVELAND, OH 44114			EXAMINER KANG, INSUN	
			ART UNIT 2193	PAPER NUMBER

DATE MAILED: 08/24/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

# Office Action Summary

Application No.

09/842,527

Applicant(s)

ANDERSON ET AL.

Examiner

Insun Kang

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 08 June 2006.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-49 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-49 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- |   |   |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)             | 4) <input type="checkbox"/> Interview Summary (PTO-413)                     |
| 2) <input type="checkbox"/> Notice of Draftperson's Patent Drawing Review (PTO-948)     | Paper No(s)/Mail Date. _____  |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____   | 6) <input type="checkbox"/> Other: _____                                    |

### **DETAILED ACTION**

1. This action is in response to the amendment filed 6/8/2006.
2. As per applicant's request, claims 41 and 42 have been amended.
3. Claims 1-49 are pending in the application.

### ***Claim Rejections - 35 USC § 101***

4. Applicant's argument is persuasive, therefore, the rejection to claims 36-39, 43-46, and 48-49 has been withdrawn.

### ***Claim Rejections - 35 USC § 103***

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. Claims 1-49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gustafsson et al. (US Patent 6,067,413) hereinafter referred to as "Gustafsson" in view of Dyer ("Java Decompiles compared," Java World, 7/1997).

Per claim 1:

Gustafsson discloses:

- a language-neutral representation of a compile unit transformable to at least one of a plurality of different types of code representations (Gustafsson, i.e. col. 2 lines 39-59)

Art Unit: 2193

- a hierarchal arrangement of program elements that neutrally characterize the compile unit (i.e. "col. 2 lines 39-59)
- at least one of the program elements representing a type declaration that characterizes at least one class of programmatic constructs of the compile unit (i.e. col. 2 lines 39-59).

Gustafsson does not explicitly teach converting the language neutral representation of the compile unit to a corresponding representation of the compile unit in at least one high-level language code. However, Dyer teaches that such decompiling method was known in the art of software development, at the time applicant's invention was made, to convert a compiled code into the original source code (page 1). It would have been obvious for one having ordinary skill in the art of computer software development to modify Gustafsson's disclosed system to use a decompiler. The modification would be obvious because one having ordinary skill in the art would be motivated to "reverse-engineer compiled code (page 1, summary)" as suggested by Dyer.

Per claim 2:

The rejection of claim 1 is incorporated, and further, Gustafsson discloses:

- a collection of at least one member that characterizes programmatic attributes associated with and able to be implemented within the at least one class (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Art Unit: 2193

Per claim 3:

The rejection of claim 2 is incorporated, and further, Gustafsson discloses an expression class within the at least one class (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 4:

The rejection of claim 2 is incorporated, and further, Gustafsson discloses a statement class within the at least one class (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 5:

The rejection of claim 2 is incorporated, and further, Gustafsson discloses a namespace that contains the at least one class (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 6:

The rejection of claim 1 is incorporated, and further, Gustafsson discloses at least one of the program elements of the hierarchal arrangement encapsulates another of the program elements (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 7:

The rejection of claim 1 is incorporated, and further, Gustafsson discloses the interface being operative to enable transformation of the language-neutral representation to a corresponding desired code representation (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 8:

Art Unit: 2193

The rejection of claim 7 is incorporated, and further, Gustafsson discloses that the program elements comprise objects, each object exposing at least one of a method, attribute, and property of each respective object, the interface being operative to employ the at least one of method, attribute and property to facilitate the transformation into the desired code representation (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 9:

The rejection of claim 7 is incorporated, and further, Gustafsson discloses a compiler interface programmed to enable transformation of the language-neutral representation to a corresponding representation in a low-level language code (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 10:

The rejection of claim 9 is incorporated, and further, Gustafsson discloses an assembly of computer-executable instructions (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 11:

The rejection of claim 7 is incorporated, and further, Gustafsson discloses a code generator interface programmed to enable conversion of the language-neutral representation to a plurality of corresponding representations, wherein each representation is in a different high-level language code (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Art Unit: 2193

Per claim 12:

The rejection of claim 1 is incorporated, and further, Gustafsson discloses

-the program elements comprise instances of a plurality of language-neutral classes, each instance defining an associated object (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 13:

The rejection of claim 12 is incorporated, and further, Gustafsson discloses

at least one associated object represents the type declaration, at least another object being encapsulated within the at least one object representing the at least one type declaration, the at least another object representing program code of the compile unit that derives from a class associated with the at least type declaration (Gustafsson, i.e. col. 3 lines 38-56) as claimed.

Per claim 14:

Gustafsson discloses:

-A language-neutral representation of compile unit (Gustafsson, i.e. col. 3 lines 38-56) - an instance of at least one of a plurality of language-neutral classes, the plurality of classes representing different programmatic constructs of a compile unit and having a hierarchal relationship relative to each other, whereby transformation of the instance into a different representation of the respective programmatic construct is facilitated (col. 3 lines 38-56)

Art Unit: 2193

Gustafsson does not explicitly teach converting the instance of ...language neutral ... to a corresponding ...representation ... in at least one high-level language code. However, Dyer teaches that such decompiling method was known in the art of software development, at the time applicant's invention was made, to convert a compiled code into the original source code (page 1). It would have been obvious for one having ordinary skill in the art of computer software development to modify Gustafsson's disclosed system to use a decompiler. The modification would be obvious because one having ordinary skill in the art would be motivated to "reverse-engineer compiled code (page 1, summary)" as suggested by Dyer.

Per claim 15:

The rejection of claim 14 is incorporated, and further, Gustafsson discloses -each instance of a corresponding class of the plurality of classes represents a respective programmatic construct of the compile unit, the plurality of instances being organized in a hierarchal relationship based on the classes associated with the plurality of instances and relationships among the programmatic constructs represented thereby (Gustafsson, i.e col. 3 lines 38-56).

Per claim 16:

The rejection of claim 15 is incorporated, and further, Gustafsson discloses that each of the plurality of instances exposes at least one item associated with the programmatic construct represented thereby (Gustafsson, i.e col. 3 lines 38-56).

Per claim 17:



Art Unit: 2193

The rejection of claim 16 is incorporated, and further, Gustafsson discloses

- at least one of the plurality of instances represents a type declaration, at least another instance being encapsulated within the instance representing the type declaration, the at least another instance representing a programmatic construct that derives from the at least type declaration(Gustafsson, i.e col. 3 lines 38-56).

Per claim 18:

The rejection of claim 17 is incorporated, and further, Gustafsson discloses

at least one of a statement and an expression(Gustafsson, i.e col. 3 lines 38-56).

Per claim 19:

The rejection of claim 16 is incorporated, and further, Gustafsson discloses an interface that enables transformation of the language neutral representation to the different representation, the interface employs the at least one item to facilitate the transformation of the language-neutral representation into the different representation (Gustafsson, i.e col. 3 lines 38-56).

Per claim 20:

The rejection of claim 19 is incorporated, and further, Gustafsson discloses a compiler interface programmed to enable transformation of the language-neutral representation to the corresponding different representation in a low-level language code (Gustafsson, i.e col. 3 lines 38-56).

Per claim 21:

The rejection of claim 20 is incorporated, and further, Gustafsson discloses an assembly of computer-executable instructions (Gustafsson, i.e col. 3 lines 38-56).

Per claim 22:

The rejection of claim 19 is incorporated, and further, Gustafsson discloses a code generator interface programmed to generate a plurality of corresponding representations from the language-neutral representation, wherein each representation is in a different high-level language code from the language-neutral representation (Gustafsson, i.e col. 3 lines 38-56).

Per claim 23:

Gustafsson discloses:

- A language-neutral representation of ... transformable to at least one other type of software code representation (Gustafsson, i.e col. 3 lines 38-56).
- a hierarchal arrangement of objects, each object representing a different program element of the compile unit class (Gustafsson, i.e col. 3 lines 38-56).
- at least one class object that represents at least one defined class of program elements of the compile unit; at least one member object associated with the at least one class object that represents computer-executable instructions operable on at least

Art Unit: 2193

some program elements in the at least one defined class(Gustafsson, i.e col. 3 lines 38-56).

Gustafsson does not explicitly teach converting the language neutral representation of computer executable instructions into a corresponding representation in at least one high-level language code. However, Dyer teaches that such decompiling method was known in the art of software development, at the time applicant's invention was made, to convert a compiled code into the original source code (page 1). It would have been obvious for one having ordinary skill in the art of computer software development to modify Gustafsson's disclosed system to use a decompiler. The modification would be obvious because one having ordinary skill in the art would be motivated to "reverse-engineer compiled code (page 1, summary)" as suggested by Dyer.

Per claim 24:

The rejection of claim 23 is incorporated, and further, Gustafsson discloses a namespace object that represents a namespace of the compile unit, the namespace object comprising a collection of class objects including the at least one class object (Gustafsson, i.e col. 3 lines 38-56).

Per claim 25:

The rejection of claim 24 is incorporated, and further, Gustafsson discloses a plurality of member objects associated with the at least one class object, wherein the at least one class object represents a common base class that is shared by the plurality of member objects (Gustafsson, i.e col. 3 lines 38-56).

Per claims 26-31, these claims are another versions of the claimed representation discussed in claims 18-22, respectively, and are rejected for the same reasons set forth in connection with the rejections of claims 18-22 above.

Per claims 32-35, these claims are another versions of the claimed representation discussed in claims 18-22, respectively, and are rejected for the same reasons set forth in connection with the rejections of claims 18-22 above.

Per claims 36-39, these claims are system versions of claims 23-26 and 31, respectively, and are rejected for the same reasons set forth in connection with the rejections of claims 23-26 above.

Per claim 40, it is the computer-readable medium version of claim 14, respectively, and is rejected for the same reasons set forth in connection with the rejection of claim 14 above.

Per claim 41, 42, it is the method version of claim 39, respectively, and is rejected for the same reasons set forth in connection with the rejection of claim 39 above. For the limitations, "the second high level language code is in a different high level language than the first high level language code," in claim 41 and "a high-level language code that is the same as the representation of the compile unit in the first high level language code" in claim 42, Gusfaffson in view of Dyer further discloses these limitations (col. 3 lines 40-56).

Per claims 43-45 and 46, these claims are system versions of claims 37 and 38 respectively, and are rejected for the same reasons set forth in connection with the rejections of claims 37 and 38 above.

Per claim 47, it is the method version of claim 29, respectively, and is rejected for the same reasons set forth in connection with the rejection of claim 29 above.

Per claim 48 and 49, it is the system version of claim 30, respectively, and is rejected for the same reasons set forth in connection with the rejection of claim 30 above.

### ***Response to Arguments***

7. Applicant's arguments filed 6/8/2006 have been fully considered but they are not persuasive.

Per claims 1, 14, 23, 32, 36, 40, 43, 47, and 48:

The Applicant states that:

Gustafsson ...discloses compiling a plurality of high level languages into a common runtime representation. The common representation is not language neutral. Rather, it is specific to one of the high level languages and the compilers for the other high level languages are modified to produce it (page 14).

In response, Gustafsson teaches the use of a shared object model enabling compilers to define their respective internal data representations in terms common to all (col. 2 lines 30-39)." The common runtime representations are considered as language-neutral representations. The instant claims do not specifically recite what the language

Art Unit: 2193

neutral representation of the compile unit is. If applicant means anything more, this must be brought out in the claims to further clarify the invention.

The applicant further states that: Dyer ...discusses examples of Java decompiles. Specifically, conversion of Java class files to Java source code...Dyer does not disclose converting a language neutral representation of a compile unit into at least one high level language (page 15)."

In response, Dyer discloses a decompiler which converts a compiled code into a source code. Gustafsson specifically states that Java compilers are modified to implement the shared symbol table (col. 3 lines 38-56). Therefore, Dyer's Java code compiler can define its respective internal data representation in terms common to all (Gustafsson, col. 2 lines 39-50) and Dyer's decompiler is applicable into Gustafsson to "reverse-engineer compiled code (Dyer, page 1, summary)."

The applicant further states that: Dyer only discusses a single high level language, Java, and conversion of...Java source code. Gustafsson et al. and Dyer fail to disclose a language neutral representation...into a high level language code (page 15).

In response, in Gustafsson, the Java compiler conforms to the internal data representation stored in the shared symbol table of the C++ compiler (col. 3 lines 40-56). The decompiler of Dyer can convert the common representation derived from C++ into Java code.

### ***Conclusion***

**8. THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

**9.** Any inquiry concerning this communication or earlier communications from the examiner should be directed to Insun Kang whose telephone number is 571-272-3724. The examiner can normally be reached on M-F 9:30-6.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on 571-272-3719. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should

Art Unit: 2193

you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

I. Kang  
AU2193

  
**KAKALI CHAKI**  
**SUPERVISORY PATENT EXAMINER**  
**TECHNOLOGY CENTER 2100**